World Academy of Science, Engineering and Technology
International Journal of Mathematical and Computational Sciences
Vol:11, No:4, 2017

Sensitivity Analysis during the Optimization Process Using Genetic Algorithms

M. A. Rubio, A. Urquia

***Abstract*—**Genetic algorithms (GA) are applied to the solution of high-dimensional optimization problems. Additionally, sensitivity analysis (SA) is usually carried out to determine the effect on optimal solutions of changes in parameter values of the objective function. These two analyses (i.e., optimization and sensitivity analysis) are computationally intensive when applied to high-dimensional functions. The approach presented in this paper consists in performing the SA during the GA execution, by statistically analyzing the data obtained of running the GA. The advantage is that in this case SA does not involve making additional evaluations of the objective function and, consequently, this proposed approach requires less computational effort than conducting optimization and SA in two consecutive steps.

***Keywords*—**Optimization, sensitivity, genetic algorithms, model calibration.

I. Introduction

OPTIMIZATION problems arise in many technical, economic and scientific projects [1]–[3]. In general, an optimization problem requires finding a setting $\vec{x} \in M$ of free parameters of the system under consideration, such that a certain quality criterion $f : M \to \Re$, called the *objective function*, is minimized (or, equivalently, maximized) [4]–[7]. The solution to this global optimization problem requires finding a vector $\vec{x}^*$ such that $\forall \vec{x} \in M : f(\vec{x}) \geq f(\vec{x}^*) = f^*$.

Any solution of the problem is represented by an array of time-independent variables, $\vec{x} = (p_1, \ldots, p_N)$, called *free parameters*, M is the search domain and f is the objective function $f : M \to \Re$.

Genetic algorithms (GA) have been successfully applied to the solution of model calibration problems [8]. The general evolutionary algorithm described in [5]–[7] is used to illustrate the application of GA to model calibration.

$Algorithm\ 1:$

$t := 0;$
initialize $P(t)$;
evaluate $P(t)$;
while not $terminate$ do
  $P'(t) :=$ variation $[P(t)]$;
  evaluate $[P'(t)]$;
  $P(t+1) :=$ select $[P'(t) \cup Q]$;
  $t := t + 1;$
od

The algorithm starts with an initial population, which may be randomly selected from the search space. $P(t) = \{\vec{x}_1(t), \ldots, \vec{x}_\mu(t)\}$ denotes a population of μ individuals at generation t. Each individual of the population corresponds to a particular value of the free parameters, $\vec{x}_k(t) = (p_{1,k}(t), \ldots, p_{N,k}(t))$, and is a candidate solution to the problem. $p_{i,k}(t)$ denotes the value that the i-th free parameter has in the k-th individual at the generation t. The number of free parameters is N. If binary coding is employed, free parameter values are represented by binary words of length n bits.

Each member of the population is evaluated replacing in the model the parameter values associated to this member and simulating the model. The objective function is used to assess the fitness of the simulated response to the experimental data. An offspring population $P'(t)$ of size λ is generated by means of variation operators (e.g., recombination and mutation) from the population $P(t)$. The offspring individuals are then evaluated and selection based on these fitness values is performed. Q is a special set of individuals that might be considered for selection (e.g., $Q = P(t)$ or $Q = \emptyset$).

As each individual evaluation implies performing a problem evaluation, the application of this technique to high dimensionality problems may be an extremely time-consuming process. For this reason, some strategies have been proposed to reduce the order of the model before solving the optimization problem. For instance, GA are used in [9] to reduce the number of free parameters in a dynamic model that describes a large reaction network. In [10] the GA was applied to reduce the search space of the classification processes. The use of sensitivity analysis (SA) techniques is reported in [11]–[13]. SA was combined with GA in order to improve the optimization process [14], [15]. SA is applied in [16] to reduce the number of free parameters in an electrochemical model, prior to its calibration using GA.

Other authors have proposed combining local SA and GA, in order to improve the robustness of the GA [17]. Local SA using parallel layer perceptron has been employed in [18] to improve the local search process of the GA.

The SA estimates the effect of free parameter variation on the objective function. Those free parameters whose variation has an small effect on the objective function value might be removed from the optimization problem. Then, the objective function might be simplified in order to reduce the

M. A. Rubio is with Departamento de Informática y Automática, E.T.S. de Ingeniería Informática, Universidad Nacional de Educación a Distancia (UNED), Juan del Rosal 16, 28040, Madrid, Spain (e-mail: marubio@dia.uned.es).

A. Urquia is with Departamento de Informática y Automática, E.T.S. de Ingeniería Informática, Universidad Nacional de Educación a Distancia (UNED), Juan del Rosal 16, 28040, Madrid, Spain.

computational effort.

When the SA is performed before solving the optimization problem, global SA methods (e.g., scatter plots and Monte Carlo [12], [13]) need to be applied to inspect the complete search space, which implies evaluating a large number of population members. This process may also be extremely time-consuming for high dimensionality problems.

The method presented in this paper, that consists in combining the SA and the GA, is aimed to reduce the computational cost. The global SA at the beginning, and the local SA when the GA is converging to the optimum, are performed during the execution of the GA, using the data obtained of running the GA. Simplification of the objective function at different levels of locality is supported.

II. Genetic Algorithm and Sensitivity Analysis

The presented algorithm (see Algorithm 2) is aimed to solve the model calibration problem described in Section I. Based on Algorithm 1, Algorithm 2 contains an additional step, $P'(t) :=$ GA-SA $[P(t)]$, in which the SA is performed.

Algorithm 2 :

$t := 0$;
initialize $P(t)$;
evaluate $P(t)$;
while not $terminate$ do
  $P'(t) :=$ GA-SA $[P(t)]$;
  $P''(t) :=$ variation $[P'(t)]$;
  evaluate $[P''(t)]$;
  $P(t+1) :=$ select $[P''(t) \cup Q]$;
  $t := t + 1$;
od

A. Initial Population

Algorithm 2 starts with an initial population composed of μ individuals, $P(0) = \{\vec{x}_1(0), \ldots, \vec{x}_\mu(0)\}$. The initial values of any free parameter p_i (i.e., $\{p_{i,1}(0), \ldots, p_{i,\mu}(0)\}$) are independent variates, uniformly distributed over the parameter range. As will be discussed in Section II-B, this initialization procedure is the basis for the presented SA method.

B. The Idea behind the Sensitivity Analysis Procedure

The Schema Theorem states that the number of representatives of the schema H in the population at $t+1$, $m(H, t+1)$, can be estimated from (1), where $m(H,t)$ is the number of representatives at t, $f(H)$ is the average fitness of the schema and $\overline{f}$ is the average fitness of the population.

$$m(H, t+1) = m(H,t)\frac{f(H)}{\overline{f}} \tag{1}$$

Without loss of generality, let's suppose that the parameter values are represented by n-bit words. Individuals are defined by strings of $n \cdot N$ bits, where N is the number of free parameters. The $H_{p_i=b}$ schema, with b a n-bit number, represents all individuals of the search space that satisfy $p_i = b$. The schema order (i.e., the number of fixed positions) is n. The search space region defined by $H_{p_i=b}$ contains $2^{n\cdot(N-1)}$ individuals.

If the objective function is insensitive to the free parameter p_i, then $f(H_{p_i=b}) = \overline{f}$. This implies $m(H_{p_i=b}, t+1) = m(H_{p_i=b}, t)$. Consequently, the distribution of the values $\{p_{i,1}(t), \ldots, p_{i,\mu}(t)\}$ will not change in time. As the values of p_i in the initial population are uniformly distributed over the parameter range, the values of p_i will be uniformly distributed in successive generations.

On the contrary, if the objective function is sensitive to the free parameter p_i, then the average fitness of $H_{p_i=b}$ depends on the value of b. Schemata with fitness values above the population average will receive an increasing number of samples in the next generation, while schemata with fitness values below the population average will receive a decreasing number of samples. Consequently, the distribution of p_i values, $\{p_{i,1}(t), \ldots, p_{i,\mu}(t)\}$, will depart from the initial uniform distribution as the GA progresses.

III. Some Methods for Sensitivity Analysis in GA-SA

As discussed in the previous section, the goal of the sensitivity analysis (SA) is to ascertain how well the parameter values resemble uniform random variates. A variety of methods can be used to this end, including the empirical tests for random number generators. Five methods are discussed in this section: Kolmogorov-Smirnov, chi-square, histogram bin size, deviation from the generation optimum and variance analysis. Different implementations of the Algorithm 2, supporting these methods in the GA-SA step, have been programmed and applied to solve three different optimization problems. The GA executions are analyzed to compare the performance of the five methods.

A. Kolmogorov-Smirnov Test

Kolmogorov-Smirnov (K-S) tests [19] compare an empirical distribution function with the distribution function of the hypothesized distribution. In this application, the i-th free parameter values at generation t, $\{p_{i,1}(t), \ldots, p_{i,\mu}(t)\}$ need to be transformed from the parameter search range into the unity interval, $[0, 1]$. Let $\{\tilde{p}_1, \ldots, \tilde{p}_\mu\}$ be the transformed values. The K-S test is used to compare the empirical distribution function of these transformed values with the uniform distribution U(0,1).

The K-S test statistic, D_μ, is the largest vertical distance between the empirical and the hypothesized distribution functions. In this application, it can be computed by calculating

$$D_\mu^+ = \max_{1\le k\le \mu}\left\{\frac{k}{\mu} - \tilde{p}_k\right\} \tag{2}$$

$$D_\mu^- = \max_{1\le k\le \mu}\left\{\tilde{p}_k - \frac{k-1}{\mu}\right\} \tag{3}$$

and finally letting

$$D_\mu = \max\left\{D_\mu^+, D_\mu^-\right\} \tag{4}$$

The null hypothesis is rejected if D_μ exceeds the value of the test critical point, $d_{\mu,1-\alpha}$, where α is the specified level of the test.

B. Chi-Square Test

The chi-square test [20] can be used to check whether the values of the i-th free parameter at generation t, $\{p_{i,1}(t), \ldots, p_{i,\mu}(t)\}$, appear to be uniformly distributed. The parameter values need to be transformed from the parameter search range into the unity interval. The $[0, 1]$ interval is divided into N_{bin} subintervals of equal length. For $m = 1, \ldots, N_{bin}$, let s_m be the number of values that are in the m-th subinterval, and let

$$\chi^2 = \frac{N_{bin}}{\mu} \sum_{m=1}^{N_{bin}} \left(s_m - \frac{\mu}{N_{bin}}\right)^2 \tag{5}$$

Then for large μ, χ^2 will have an approximate chi-square distribution with $N_{bin} - 1$ degrees of freedom, under the null hypothesis that the free parameter values, normalized to the unity interval, are independent, identically distributed U(0,1) random variables.

C. Histogram Bin Size

Histograms show what proportion of data falls into each interval of the parameter range. Many methods have been proposed for selecting the bin size of the histogram. According to the method described in [21], the optimum bin size is the value of Δ that minimizes $c(\Delta)$ in (6).

$$c(\Delta)_i = \frac{2 \cdot \overline{s}_i(t) - \mu \cdot \chi_i^2(t)}{\Delta^2} \tag{6}$$

If the data are independent variates distributed $U(0, R)$, then $\chi_i^2(t) \cong 0$, $\overline{s}_i(t) = \mu/N_{bin}$ and $\Delta = R/N_{bin}$, where μ is the number of data. Equation (7) is obtained replacing these values in (6).

$$c(N_{bin}) = \frac{2 \cdot \mu \cdot N_{bin}}{R^2} \tag{7}$$

The minimum value of $c(N_{bin})$ in (7) corresponds to $N_{bin} = 1$ or, equivalently, to $\Delta = R$. The optimum number of bins for independent variates obtained from a uniform distribution $U(0, R)$ is one and the minimum value of c is given by (8).

$$c_{min} = \frac{2 \cdot \mu}{R^2} \tag{8}$$

This result can be used to quantify the departure of a data set from the uniform distribution. The optimum number of bins is calculated for the data set, minimizing $c(\Delta)$ in (6). The larger the difference between the calculated minimum and $2 \cdot \mu/R^2$, the larger the departure of the data distribution from $U(0, R)$.

D. Deviation from the Generation Optimum

The individual with the best fitness of the population $P(t)$ is represented as $\vec{x}^*(t)$. His parameter values are represented as $\vec{x}^*(t) = (p_1^*(t), \ldots, p_N^*(t))$. The average distance of the population members with respect to the individual with best fitness can be calculated, for each parameter p_i^*, from (9). This statistical indicator is used to establish a comparison among the sensitivity to different parameters. This method does not provide a reference value for the uniform distribution.

$$d_i(t) = \frac{1}{\mu} \sum_{k:1}^{\mu} |p_i^*(t) - p_{i,k}(t)| \qquad \text{for } i : 1, \ldots, N \tag{9}$$

E. Variance Analysis

The normalized standard deviation of a random variable is defined as its standard deviation divided by its range. An estimation of the normalized standard deviation of each parameter $p_i(t)$ can be calculated at time t from the data $\{p_{i,1}(t), \ldots, p_{i,\mu}(t)\}$. This calculated value can be compared with $1/\sqrt{12}$, which is the normalized standard deviation of a uniform distribution $U(a, b)$. A drawback of this method is that it can erroneously indicate that the objective function is insensitive to a parameter, when in fact it is sensitive. This is the case if the parameter values are grouped around the minimums of a symmetric objective function and the calculated normalized standard deviation is close to the reference value for the uniform distribution, i.e., $1/\sqrt{12}$.

F. Comparison of the Five Methods

The three objective functions described below will be used to evaluate the five methods for sensitivity analysis discussed in Sections III-A-III-E.

– De Jong's function (monomodal function with separated variables) [22].

$$f_1 = \sum_{i:1}^{N} A_i \cdot p_i^2 \tag{10}$$

– Rastringin's function (multimodal function with separated variables) [23].

$$f_2 = 10 \cdot N + \sum_{i:1}^{N} A_i \cdot (p_i^2 - 10 \cdot \cos(2 \cdot \pi \cdot p_i)) \tag{11}$$

– Griewangk's function (multimodal function with non-separable variables) [24].

$$f_3 = 1 + \frac{1}{4000} \sum_{i:1}^{N} A_i \cdot p_i^2 - \prod_{i:1}^{N} \cos(\frac{A_i \cdot p_i}{\sqrt{i}}) \tag{12}$$

Each objective function has $N = 6$ free parameters: $\{p_1, \cdots, p_6\}$. The optimization problem consists in finding the value of these parameters that minimizes the function. The search range of each free parameters is $[-5.12, 5.11]$.

A sensitivity vector A of six binary components (i.e., of values $\{0, 1\}$) has been included in the functions. It allows

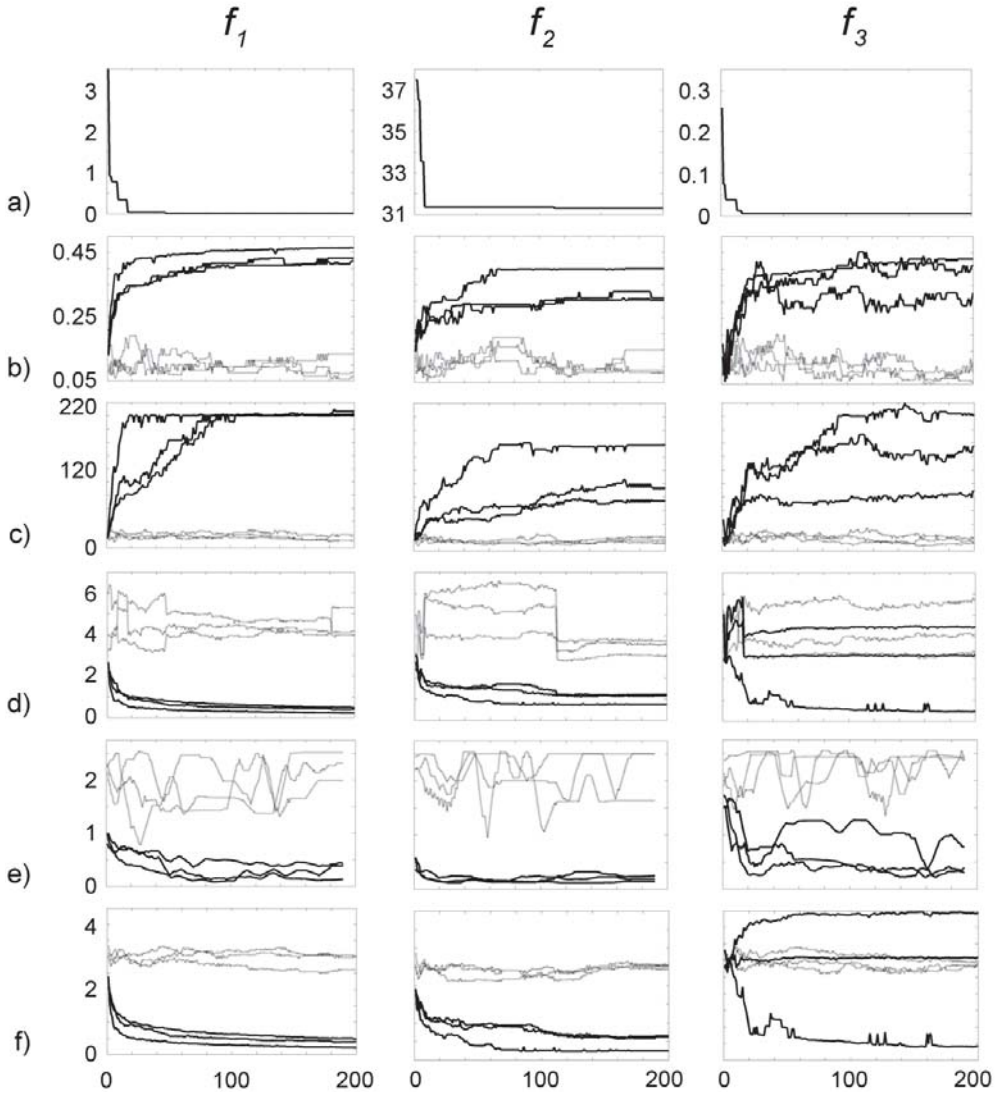

Fig. 1 Evaluation of sensitivity analysis methods using three objective functions: De Jong's (f_1), Rastringin's (f_2) and Griewangk's (f_3) function. The evolution over 200 generations of the objective function value is shown in (a)). The objective function sensitivity to the free parameters is evaluated using the following methods: (b) K-S, (c) chi-square, (d) histogram bin size, (e) deviation from the generation optimum, and (f) variance analysis. Sensitive parameters (— line): p_1, p_2 y p_4. Insensitive parameters (- - - line): p_3, p_5 and p_6

determining whether the function is sensitive or insensitive with respect to each parameter. If $A_i = 0$, the function is insensitive to p_i. On the contrary, if $A_i = 1$, the function is sensitive to p_i. The sensitivity vector is $A = (1, 1, 0, 1, 0, 0)$. Therefore, the objective function is sensitive to p_1, p_2 and p_4, and insensitive to p_3, p_5 and p_6.

The population contains $\mu = 100$ individuals. As the other individuals are randomly generated, only the 50 individuals obtained from the cross operation are considered in the sensitivity analysis.

The evolution over 200 generations of the fitness and the sensitivity metrics is shown in Fig. 1. Continuous lines are used to represent the sensitive parameters (p_1, p_2 and p_4), and dotted lines to represent the insensitive parameters (p_3, p_5 and p_6).

The K-S and chi-square methods have been successfully applied to the sensitivity analyses of the three functions. The variance analysis and the histogram bin size methods are very effective in the sensibility analysis of only f_1 and f_2.

The method based on the deviation from the generation optimum allows identifying the set of sensitive parameters of f_1 and f_2. However, this method does not provide additional information when compared with the variance analysis method and its interpretation is less straightforward, given the unavailability of a reference value for the uniform distribution.

The histogram bin size method allows analyzing correctly the sensitivity of f_3 with respect to the six free parameters. However, this method has a drawback: the statistical estimator values are noisy and they need to be filtered for facilitating their interpretation. The results shown in Fig. 1.d are the mean mobile, over a forward 10-generation window, of the original estimator values.

The intrinsic drawback of the variance analysis method,

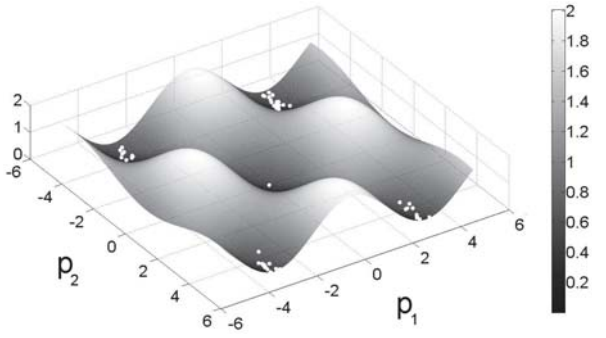

Fig. 2 p_1 and p_2 (white points) at the 100-th population of function f_3 (surface) optimization

described in Section III-E, arises during the sensitivity analysis of f_3. To illustrate it, the values of p_1 and p_2 at the 100-th population are shown in Fig. 2. The analysis indicates that f_3 is insensitive to p_1, which is wrong. The variance of the p_2 values is greater than the uniform distribution variance. As a result, the method correctly concludes that the objective function is sensitive to p_2.

IV. Conclusion

A method that combines optimization using GA, and SA has been proposed. The method is valid for any GA that satisfies the following condition: The initial population is uniformly distributed over the initial search space. The sensibility analysis is performed during the execution of the GA, by analyzing the data generated of running the GA. This SA provides valuable information at different locality levels. At the beginning of the optimization process, the SA is not focused in any value of the search space and, consequently, the SA is global. On the contrary, when the optimization process is close to finding an optimal, the SA is local around the optimum.

The advantage of the proposed method is that the sensitivity analysis does not involve additional evaluations of the objective function. Therefore, the computational effort may be significantly smaller than the traditional two-step approach, in which the sensitivity analysis and the optimization problem are performed one at a time.

Acknowledgment

This work was supported by Ministerio de Economía y Competitividad of Spain grant DPI2013-42941-R.

References

[1] C. D. Chio, A. Brabazon, M. Ebner, M. Farooq, A. Fink, J. Grahl, G. Greenfield, P. Machado, M. O'Neill, E. Tarantino, and N. Urquhart, *Applications of Evolutionary Computation*. Springer, 2010.

[2] R. L. Johnston, *Applications of Evolutionary Computation in Chemistry*. Springer, 2004.

[3] C. Karr and L. M. Freeman, *Industrial Applications of Genetic Algorithms*. CRC Press, 1998.

[4] T. Back, U. Hammel, and H.-P. Schwefel, "Evolutionary computation: Comments on the history and current state," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 3–17, 1997.

[5] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[6] L. Davis, *Handbook of Genetic Algorithms*. Van Nostrand Reinhold Company, 1991.

[7] A. E. Eiben and J. Smith, *Introduction to Evolutionary Computing*. Springer, 2010.

[8] M. A. Rubio, A. Urquia, and S. Dormido, "An approach to the calibration of Modelica models," in *Proceedings of the 1st International Workshop on Equation-Based Object-Oriented Languages and Tools*, 2007.

[9] K. Edwards, T. F. Edgar, and V. Manousiouthakis, "Kinetic model reduction using genetic algorithms," *Computers chem. Engng.*, vol. 22, pp. 239–246, 1998.

[10] M. L. Raymer, W. F. Punch, E. D. Goodman, L. A. Kuhn, and A. K. Jain, "Dimensionality reduction using genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 2, pp. 164 – 171, 2000.

[11] K. Chan, A. Saltelli, and S. Tarantola, "Sensitivity analysis of model output: variance-based methods make the difference," in *Proceedings of the Winter Simulation Conference*, 1997.

[12] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global Sensitivity Analysis: The Primer*. Wiley, 2008.

[13] A. Saltelli, K. Chan, and E. M. Scott, *Sensitivity Analysis*. Wiley, 2008.

[14] W. Chen, X. Lu, C. Yao, G. Zhu, and Z. Xu, "An efficient approach based on bi-sensitivity analysis and genetic algorithm for calibration of activated sludge models," *Chemical Engineering Journal*, vol. 259, pp. 845–853, 2015.

[15] M. Vazquez-Cruz, R. Guzman-Cruz, O. C.-P. I.L. Lopez-Cruz, I. Torres-Pacheco, and R. Guevara-Gonzalez, "Global sensitivity analysis by means of efast and sobol methods and calibration of reduced state-variable tomgro model using genetic algorithms," *Computers and Electronics in Agriculture*, vol. 100, pp. 1–12, 2014.

[16] C. Andres, C. F. Valerio, and P. Franz, "Validation of an advanced lithium-ion battery model for electric and hybrid drive trains," in *EET-2008 European Ele-Drive Conference. International Advanced Mobility Forum*, 2008.

[17] N. Nishida, Y. Takahashi, and S. Wakao, "Robust design optimization approach by combination of sensitivity analysis and sigma level estimation," *IEEE Transactions on Magnetics*, vol. 44, no. 6, pp. 998–1001, 2008.

[18] D. G. Vieira, D. A. G. Vieira, W. M. Caminhas, and J. A. Vasconcelos, "A hybrid approach combining genetic algorithm and sensitivity information extracted from a parallel layer perceptron," *IEEE Transactions on Magnetics*, vol. 41, no. 5, pp. 1740 – 1743, 2005.

[19] W. T. Eadie, *Statistical Methods in Experimental Physics*. Elsevier, 1983.

[20] R. L. Plackett, "Karl Pearson and the chi-squared test," *International Statistical Review*, vol. 51, no. 1, pp. 59–72, 1983.

[21] H. Shimazaki and S. Shinomoto, "A method for selecting the bin size of a time histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, 2007.

[22] K. A. D. Jong, "An analysis of the behavior of a class of genetic adaptive systems," Ph.D. dissertation, University of Michigan Ann Arbor, MI, USA, 1975.

[23] H. Muhlenbein, M. Schomisch, and J. Born, "The parallel genetic algorithm as function optimizer," *Parallel Computing*, vol. 17, no. 6-7, pp. 619 – 632, 1991.

[24] T. Back, D. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*. Bristol and Oxford University Press, 1997.